**Projet 2 : Détection des mots en utilisant edge impulse**

Dans ce projet, nous allons construire un modèle à l’aide d’edge impulse et la carte arduino nano 32 BLE permettant de reconnaitre les mots suivnants : « FPGA », « Résistance », « Transistor » et « microcontroleur »

Enoncé du projet : [Projets pratique TinyML - Google Docs](https://docs.google.com/document/d/165lPAe9y8E3y0lN57TPCL7ZNIMpRgXh6Yqc1J_B92Yg/edit)

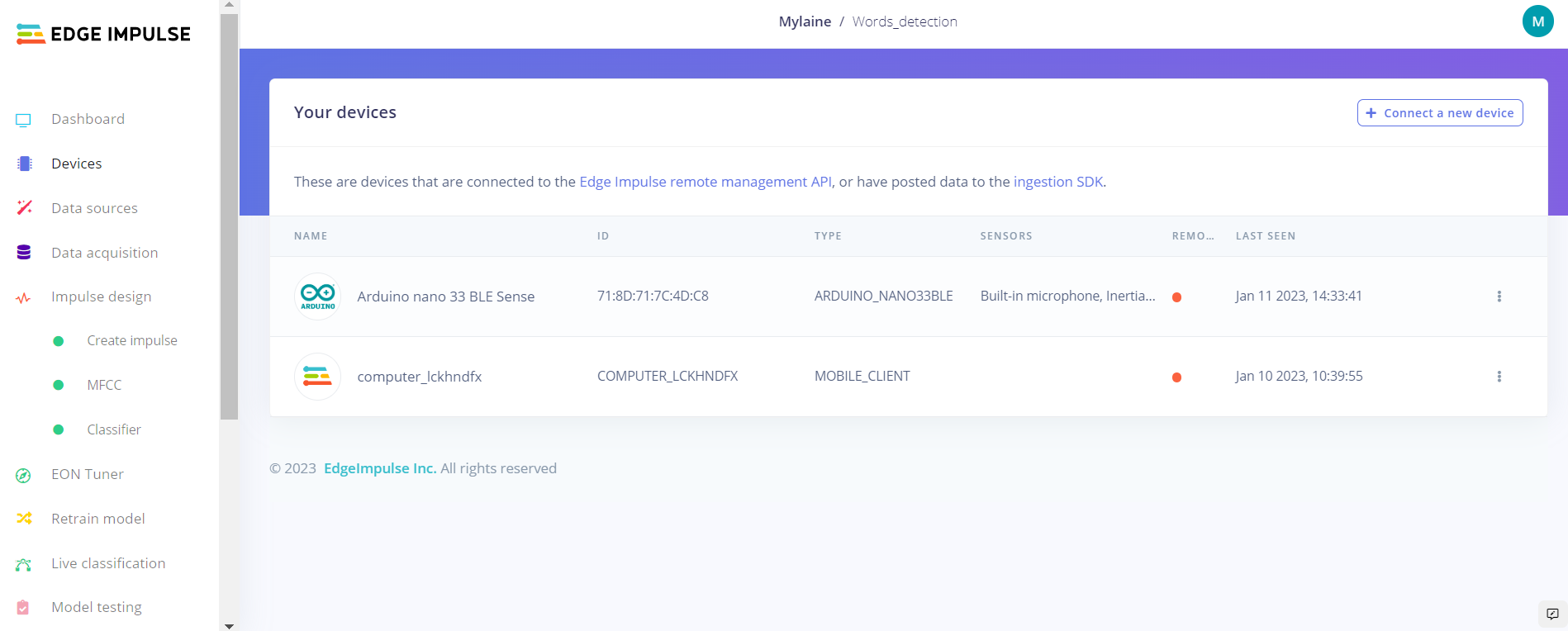
Liens utiles : [Edge Impulse with the Nano 33 BLE Sense | Arduino Documentation | Arduino Documentation](https://docs.arduino.cc/tutorials/nano-33-ble-sense/edge-impulse#creating-an-impulse)

**Connect to device :**

Nous devons dans un premier temps créer un compte sur edge impulse et ensuite connecter la carte à la plateforme.

Le lien suivant nous montre les étapes de connections d’une carte à l’interface edge impulse

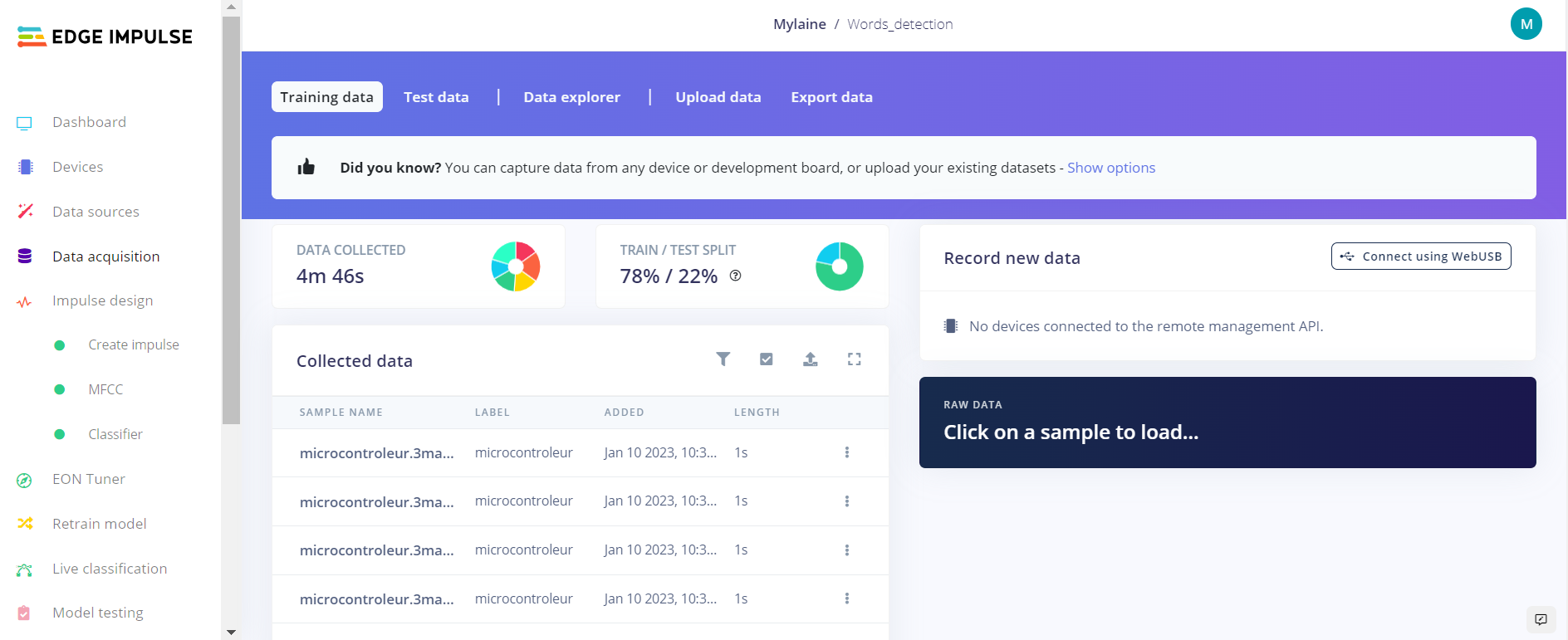
<https://docs.edgeimpulse.com/docs/development-platforms/fully-supported-development-boards>



Après s’être connecter, nous devons recueillir les donner en utilisant le microphone de la carte

* **Data acquisition :**

En utilisant le microphone de notre arduino nano BLE 33, nous devons enregistrons nos mots les uns après les autres en les catégorisant par label



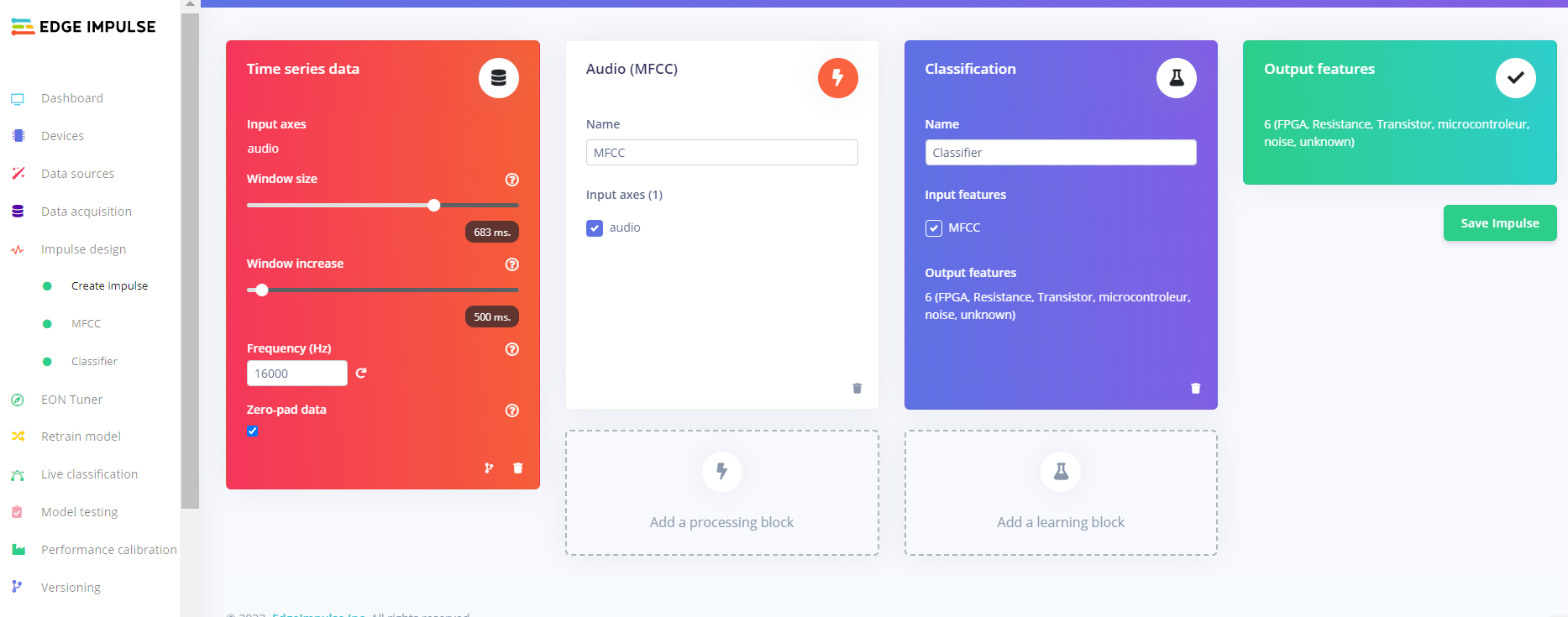
Nous devons nous rassurer que la quantité de données collectées est sensiblement la même pour tous les labels.

Avant de passer au test, nous devons diviser nos données en deux catégories (une pour l’apprentissage et l’autre pour le test).

Dashboard -> perform split/train data

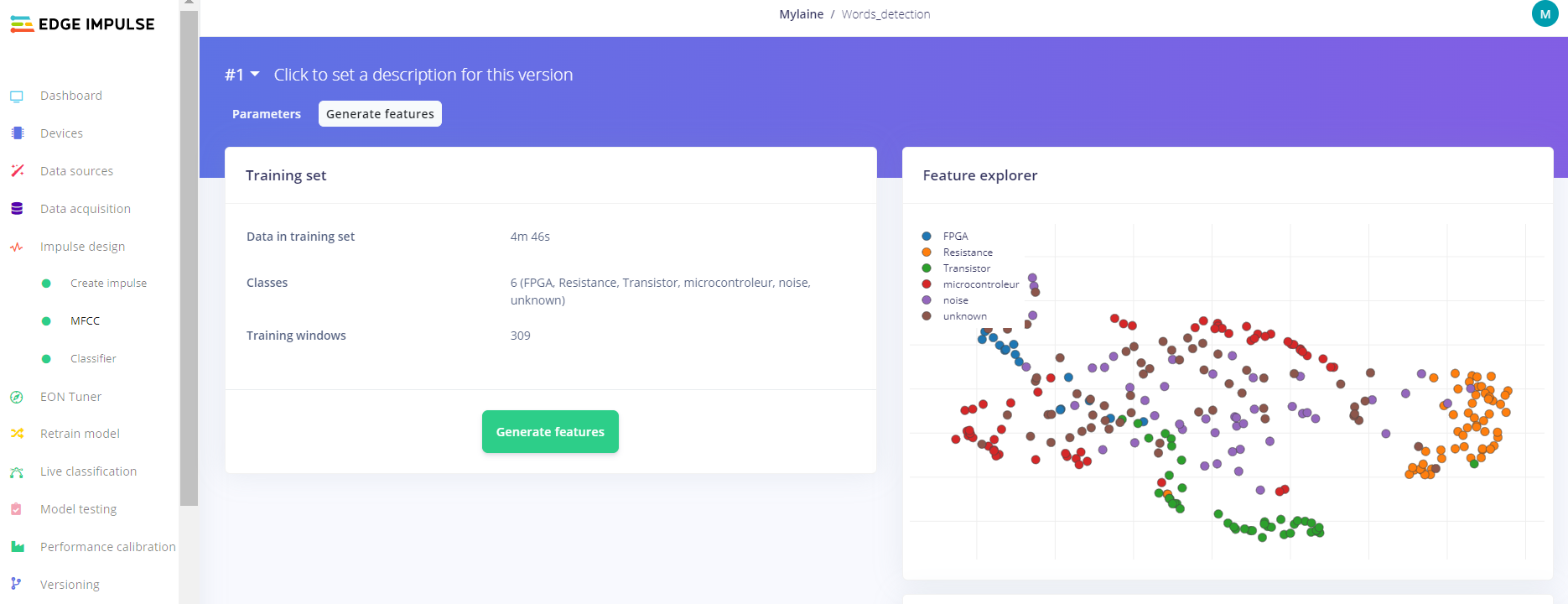
* **Create impulse :**

Choisir audio comme type de données et la classification comme méthode d’entrainement

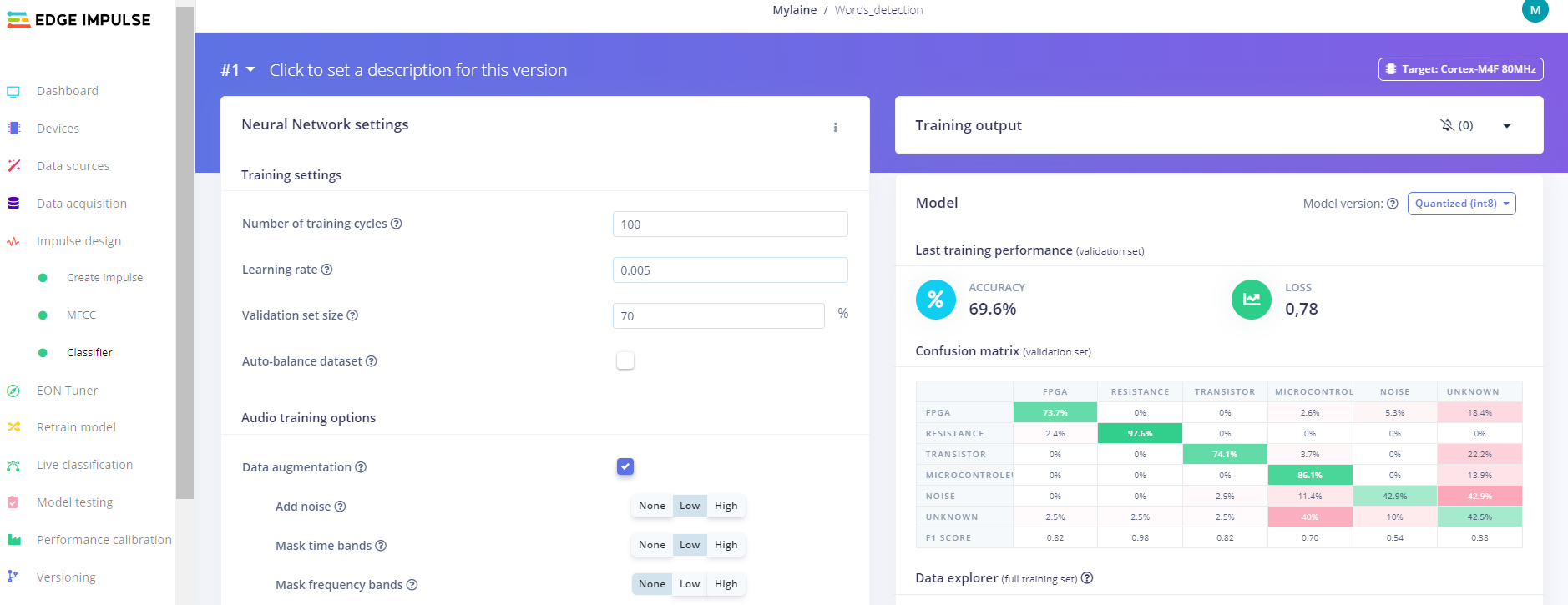


* **MCCF**

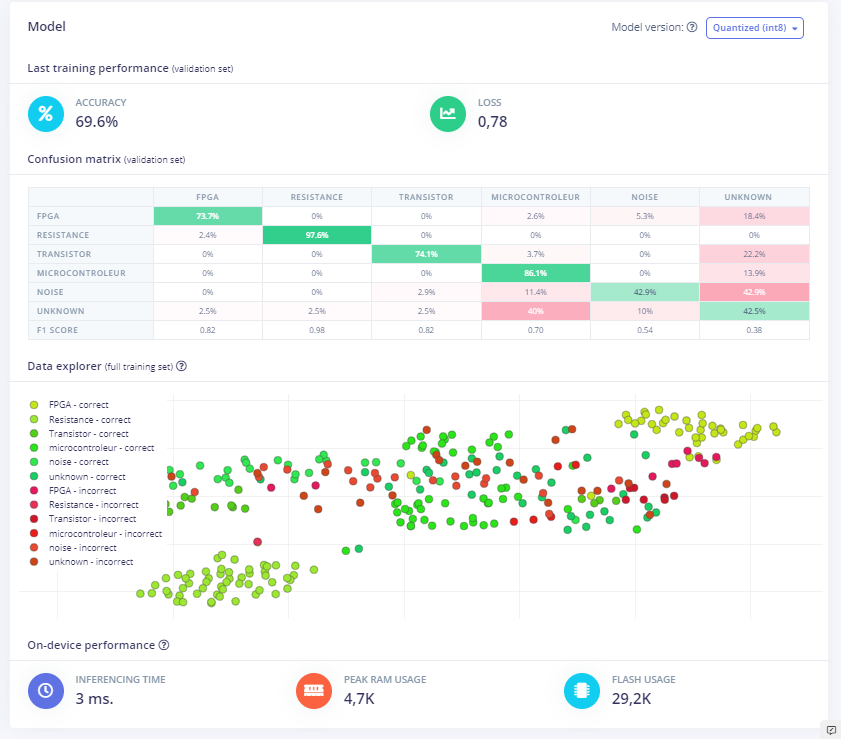
Enregistrer les paramètres -> generate features :



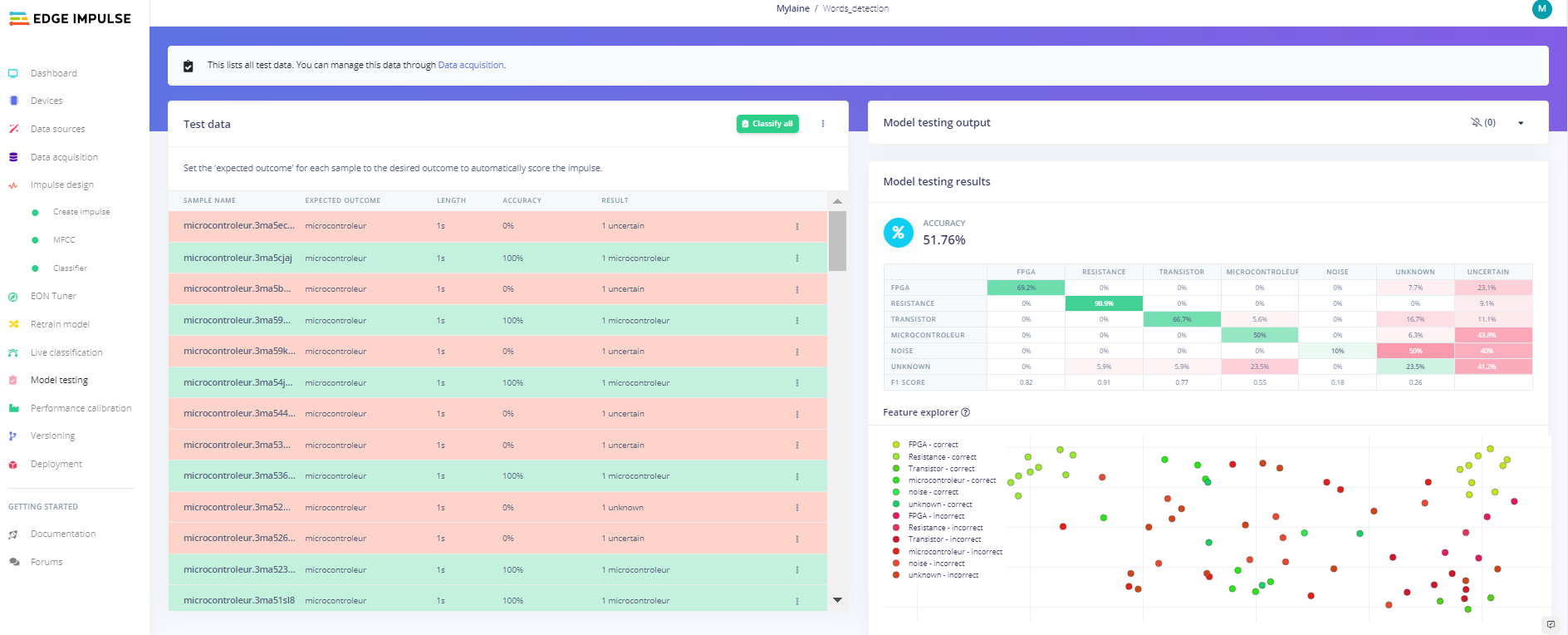
* Classifier :



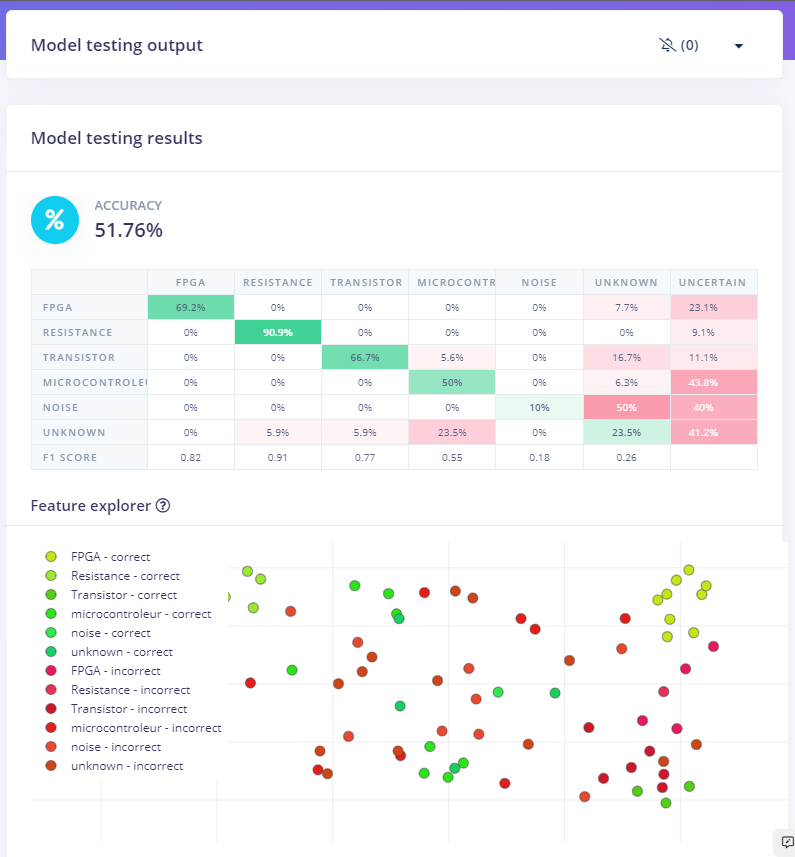
Training output :



* Model testing : cliquer sur clasiffy all

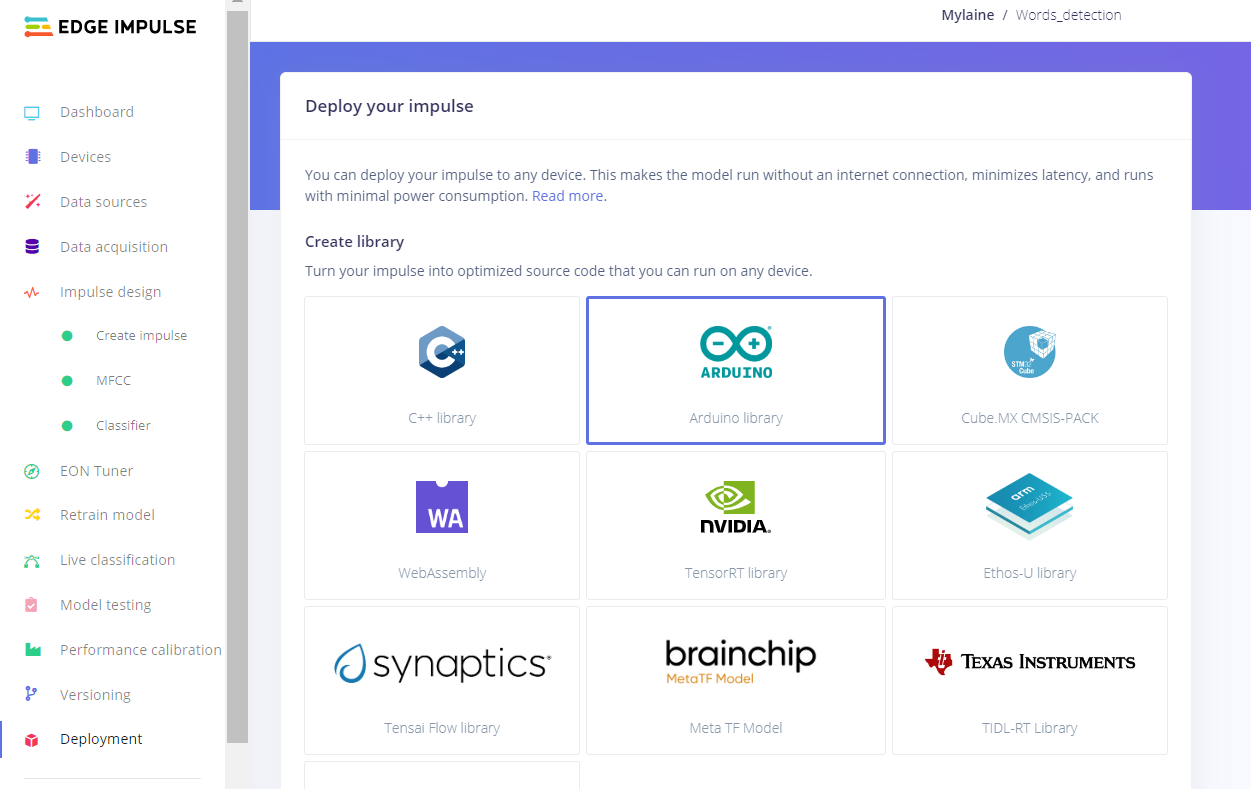


Testing output :

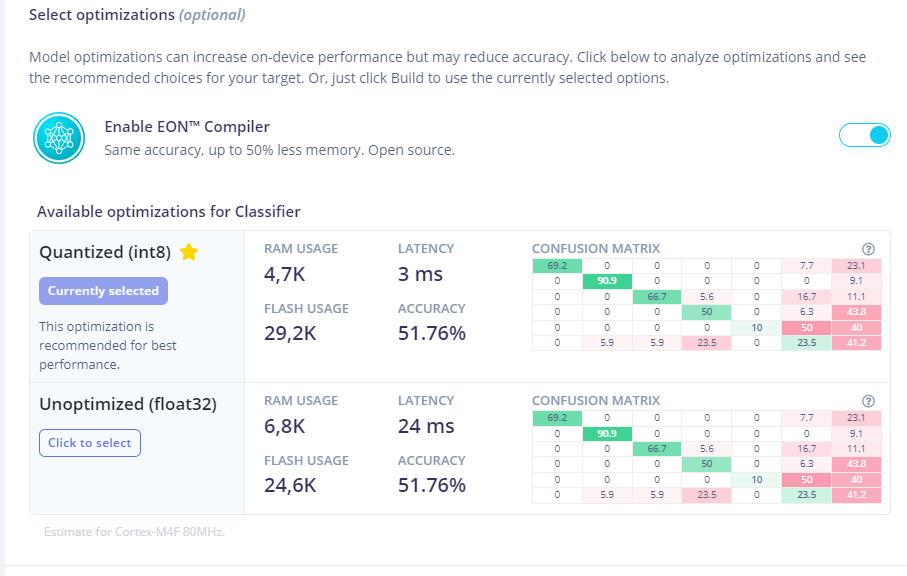


* Deployment :

Sélectionner arduino library pour une utilisation sur l’IDE arduino



Run optimization

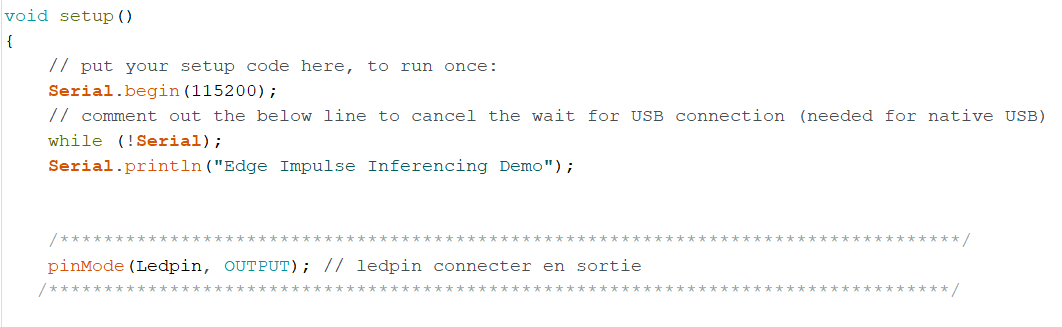


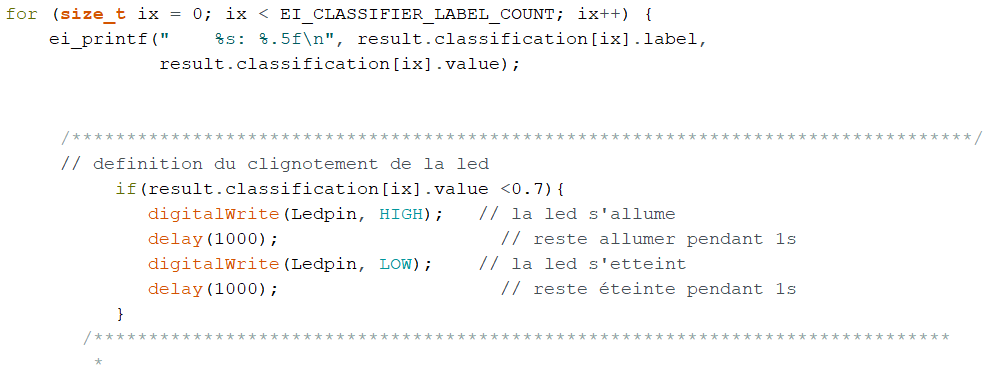
En enfin build.

Celui-ci nous génère un dossier.zip que nous devons inclure dans l’IDE arduino en tant que librairie et utiliser des exemples à nos intérêts.

Ouvrir le fichier exemple d’utilisation continue à l’aide d’un microphone et ajouter les débris de code suivant :







Dans l’onglet outils, sélectionner le type de carte et le port et enfin compiler et televerser le programme sur la carte